



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/797,515	03/10/2004	Steve Naroff	772.025US1	4252

45833 7590 05/06/2009
SCHWEGMAN, LUNDBERG & WOESSNER/APPLE
PO BOX 2938
MINNEAPOLIS, MN 55402

EXAMINER

NGUYEN, PHILLIP H

ART UNIT	PAPER NUMBER
----------	--------------

2191

MAIL DATE	DELIVERY MODE
-----------	---------------

05/06/2009

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/797,515	NAROFF ET AL.	
	Examiner	Art Unit	
	Phillip H. Nguyen	2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 17 March 2008.

2a) This action is **FINAL**. 2b) This action is non-final.

3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-45 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) Claim(s) _____ is/are allowed.

6) Claim(s) 1-45 is/are rejected.

7) Claim(s) _____ is/are objected to.

8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.

10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) All b) Some * c) None of:
1. Certified copies of the priority documents have been received.
2. Certified copies of the priority documents have been received in Application No. _____.
3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) Notice of References Cited (PTO-892)

2) Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) Information Disclosure Statement(s) (PTO/SB/08)

4) Interview Summary (PTO-413)

Paper No(s)/Mail Date. _____.

5) Notice of Informal Patent Application

6) Other: _____

DETAILED ACTION

1. This action is in response to the amendment filed 3/17/2008.
2. Per applicant's request, claims 1, 7, 12, 17, 21, 26, 30, 36, and 41 have been amended; claims 46-51 have been cancelled; claims 1-45 remain pending and have been considered below.

Response to Arguments

3. Applicant's arguments filed 3/17/2008 have been fully considered but they are not deemed persuasive.

Applicant asserts on pages 10-15 of the amendment regarding claims 1, 7, 12, 17, 21, and 26 that Szoke fails to teach *loading a first set of instructions into a execution without loading the third set of instruction into the execution unit and without loading the second set of instructions into the execution unit when loading the first set of instructions.*

Examiner respectfully disagrees with all the allegations as argued. In order to execute the second set and third set of instructions, they **MUST** be loaded into the execution unit. If the second and third sets of instructions are not loaded, they could not be executed by the execution unit. In this case, if the third set of instructions and the second set of instructions are not loaded, the unresolved reference to the second set of instructions could not be resolved by the third set of instructions and could possibly create runtime errors because the second set of instructions is not loaded.

Examiner is entitled to give claim limitations their broad reasonable interpretation in light of the specification. See MPEP 2111 [R-1] Interpretation of Claim-Broadest Reasonable Interpretation. During patent examination, the pending claims must be given their broadest reasonable interpretation consistent with the specification.

Applicant always has the opportunity to amend the claims during the prosecution and broad interpretation by the examiner reduces the possibility that the claims once issued, will be interpreted more broadly than is justified. *In re Prater*, 162 USPQ 541, 550-51 (CCPA 1969).

Specification

4. The amendment filed 3/17/2008 is objected to under 35 U.S.C. 132(a) because it introduces new matter into the disclosure. 35 U.S.C. 132(a) states that no amendment shall introduce new matter into the disclosure of the invention. The added material which is not supported by the original disclosure is as follows: "...without loading the third set of instructions into the execution unit and without loading the second set of instructions into the execution unit when loading the first set of instruction" recited in claim 1 or similar concept recited in other independent claims are not originally disclosed in the specification.

Applicant is required to cancel the new matter in the reply to this Office Action.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

6. Claims 1-45 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. "...without loading the third set of instructions into the execution unit and without loading the second set of instructions into the execution unit when loading the first set of instruction" recited in claim 1 or similar concept recited in other independent claims are not originally disclosed in the specification. All the dependent claims are suffered the same deficiency.

Claim Rejections - 35 USC § 112

7. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

8. Claims 1-45 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Regarding claims 1, 7, 12, 17, 21, 26, 30, 36, and 41 recite "...without loading the third set of instructions into the execution unit and without loading the second set of instructions into the execution unit when loading the first set of instruction" is unclear and contradicting with other claimed limitations of the claims. Instructions must be

loaded into the execution unit in order to be executed. For examining purposes, the examiner interprets that the second and third sets of instructions are loaded in order to be executed. Similar concept is also found in other independent claims. All the dependent claims are suffered the same deficiency.

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

10. Claims 1, 2, 4-8, 17, 21, 22, 24-26, 30, 31, 33-37 and 39 are rejected under 35 U.S.C. 102(b) as being anticipated by Szoke (United States Patent No.: 4,787,034).

As per claims 1 and 30:

Szoke discloses:

.- loading a first set of instructions into an execution unit (**see at least FIG. 1, “program P11”**), wherein the first set of instructions includes an unresolved reference to a second set of instructions (“**CALL P21**” or “**CALL P24**”), wherein the loading includes replacing the unresolved reference with an address of a third set of instructions (**“when the statement CALL ‘P21’ in program P11 is executed, control is passed to the statement at entry point P21 in E-Table 120, i.e., to branch statement 122. Branch**

statement 122 causes a branch to linkage program 130" col. 4, line 3-7,

also see at least FIG. 1);

- executing instructions of the first set ("when the statement CALL 'P21' in **program P11 is executed**" col. 4, line 3-4);
- executing instructions of the third set ("Branch statement 122 causes a **branch to linkage program 130**" col. 4, line 6-7), wherein executing instructions of the third set includes loading the second set of instructions into the execution unit ("when load module 200 is loaded by linkage program **130 of module 100**" col. 4, line 25-26); and
- executing instructions of the second set ("linkage program 130 transfers control to the actual address of program P21 in load module 200. **Program P21 CALLs programs P22 and P23, and then returns to program P11**" col. 4, line 45-48).

As per claims 2, 22 and 31:

Szoke discloses:

- wherein the first set of instructions includes an executable object module ("load module 100 (LM-1)" col. 2, line 53-54, **also see at least FIG. 1**).

As per claims 4, 24 and 34:

Szoke discloses:

- wherein the second set of instructions includes a separately compiled object module (“**load module 200 (LM-2)**” col. 2, line 54, **also see at least FIG. 1**).

As per claims 5, 25 and 35:

Szoke discloses:

- wherein the third set of instructions includes a loader unit (“**linkage program 130**” col. 4, line 7, **also see at least FIG. 1**).

As per claims 6 and 33:

Szoke discloses:

- wherein the loading does not include determining whether the unresolved reference refers to a defined external symbol (“**linkage program 130 determines the address of T-table 260 from the load address of load module 200 supplied by the operating system**” col. 4, line 17-19, **therefore, Szoke does not determine whether the unresolved reference refers to a defined external symbol**).

As per claims 7 and 36:

Szoke discloses:

- compiling a source code module into an executable object module that includes an unresolved reference to a separately compiled object module (“**the compiler has no way to determine the address of the callable**

program, the compiler lists the callable program as an unresolved external reference col. 1, line 38-40);

- loading the executable object module (**see at least FIG. 1, "program P11"**), wherein the loading includes replacing the unresolved reference with a reference to a system module (**"when the statement CALL 'P21' in program P11 is executed, control is passed to the statement at entry point P21 in E-Table 120, i.e., to branch statement 122. Branch statement 122 causes a branch to linkage program 130"** col. 4, line 3-7, **also see at least FIG. 1**), and wherein neither the compiling nor the loading include determining whether the unresolved reference refers to a defined external symbol (**"linkage program 130 determines the address of T-table 260 from the load address of load module 200 supplied by the operating system"** col. 4, line 17-19, **therefore, Szoke does not determine whether the unresolved reference refers to a defined external symbol**);
- executing the executable object module, wherein the executing includes, calling the system module for loading the separately compiled object module (**"when load module 100 is created (executed). Therefore, when the statement CALL 'P21' in program P11 is executed"** col. 4, line 2-4); and
- executing the separately compiled object module (**"linkage program 130 transfers control to the actual address of program P21 in load module 200. Program P21 CALLs programs P22 and P23, and then returns to program P11"** col. 4, line 45-48).

As per claims 8 and 39:

Szoke discloses:

- wherein the system module includes a loader unit (“**linkage program 130**” col. 4, line 7).

As per claim 17:

Szoke discloses an apparatus comprising:

- a compiler unit to create an executable object module based on a source code module (“**compiler**” col. 1, line 38), wherein the executable object module includes an unresolved reference to a separately compiled object module (“**the compiler lists the callable program as an unresolved external reference**” col. 1, line 39-41);
- a storage unit to store the executable object module (“**storage space**” col. 1, line 65);
- an execution unit to receive the executable object module (“**linkage editor**” col. 3, line 9); and
- a loader unit to find the executable object module in the storage unit and present the executable object module to the execution unit (“**linkage program 130 causes the operating system to load the load module identified by literal constant 123, i.e., load module 200**” col. 4, line 7-10), wherein the loader unit is to replace the unresolved reference with a

reference to a system module (“**when load module 200 has been loaded, the operating system will return the load address of load module 200 to linkage program 130**” col. 4, line 10-12), and wherein the loader unit is not to determine whether the unresolved reference refers to a defined external object module (“**linkage program 130 determines the address of T-table 260 from the load address of load module 200 supplied by the operating system**” col. 4, line 17-19, **therefore, Szoke does not determine whether the unresolved reference refers to a defined external symbol**).

As per claim 21:

Szoke discloses:

- a loader unit to load a first set of instructions into a memory unit (“**load module 100 (LM-1)**”, col. 2, line 53-54, **this means, the module is already loaded in memory**), wherein the first set of instructions includes an unresolved reference to a second set of instructions (“**when load module 100 is created the statements CALL ‘P21’ in program P11 and CALL ‘P24’ in program P12 would not ordinarily be resolved, and would instead be listed as unresolved external references**” col. 3, line 5-10), the loader unit to replace the unresolved reference with an address of a third set of instructions (“**when load module 200 has been loaded, the operating system will return the load address of load module 200 to linkage program 130**” col. 4, line 10-12); and

- an execution unit to execute instructions of the first set (“**when load module 100 is created (executed)**” col. 4, line 2), the execution unit also to execute instructions of the third set to determine an address of the second set of instructions (“**Branch statement 122 causes a branch to linkage program 130**” col. 4, line 6-7), wherein the loader unit is to use the address of the second set of instructions to load the second set of instructions into the memory unit (“**when load module 200 is loaded by linkage program 130 of load module 100, these relative addresses are converted to the actual loaded addresses of program P21 and P24 respectively**” col. 4, line 25-28).

As per claim 26:

Szoke discloses a system comprising:

- a memory unit, the memory unit including, a compiler unit to create an executable object module based on a source code module, wherein the executable object module includes a symbolic reference to a separately compiled object module (“**when load module 100 is created the statements CALL ‘P21’ in program P11 and CALL ‘P24’ in program P12 would not ordinarily be resolved, and would instead be listed as unresolved external references**” col. 3, line 5-10);
- a loader unit to present the executable object module for execution, wherein the loader unit is to replace the symbolic reference with an address to a

system module (“**when load module 200 has been loaded, the operating system will return the load address of load module 200 to linkage program 130**” col. 4, line 10-12), and wherein the loader unit is not to determine whether the symbolic reference refers to a defined external object module (“**linkage program 130 determines the address of T-table 260 from the load address of load module 200 supplied by the operating system**” col. 4, line 17-19, **therefore, Szoke does not determine whether the unresolved reference refers to a defined external symbol**); and

- a processor to receive the executable object module from the loader unit of the memory unit (**it is inherent in order to process the load modules**).

As per claim 37:

Szoke discloses:

- wherein the determining the address includes looking-up the address in a master symbol table (“**linkage program 130 determines the address of T-table 260 from the load address of load module 200 supplied by the operating system**” col. 4, line 17-19).

Claim Rejections - 35 USC § 103

11. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

12. Claims 9, 14, and 43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Szoke (United States Patent No.: 4,787,034).

As per claims 9, 14, 43:

Szoke does not explicitly discloses:

- wherein the loader unit is a dyld loader.

However, it would have been obvious to one having an ordinary skill in the art at the time the invention was made to recognize that linkage program 130 is performed dynamically (“**the CALLs to programs P21 and P24 to be resolved dynamically at the time that load module 100 is executed**” col. 3, line 11-13).

Therefore, one of ordinary skill in the art would have been motivated to use dyld loader because it is also dynamic loader and available from Apple Computer, Inc.

13. Claims 3, 11, 13, 19, 23, 28, 32, 40 and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Szoke (United States Patent No.: 4,787,034), in view of “Apple Developer Connection” Apple Computer Inc. 2001.

As per claims 3, 11, 13, 19, 23, 28, 32, 40 and 42:

Szoke does not explicitly disclose:

- wherein the executable object module is in the Mach-O object format.

However, Apple Developer Connection 2001 discloses the use of Mach-O file format. It would have been obvious to one having an ordinary skill in the art at the time the invention was made to recognize that Mach-O is a well known file format in the art for executable object code and use it in Szoke's approach for storing executable object code.

Therefore, one would have been motivated to use Mach-O file format because it provides both intermediate and final storage of machine code and data. It was designed as a flexible replacement for the BSD a.out format to be used by the compiler.

14. Claim 10, 18, 20, 27, and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Szoke (United States Patent No.: 4,787,034), in view of Tatge et al. (United States Patent No.: 5,293,630).

As per claims 10, 18, 20, 27, 29, 38:

Szoke does not explicitly disclose:

- wherein the source code module includes instructions of a dialect of the C programming language.

However, Tatge discloses an analogous method of returning a data structure from a callee function to a caller function from the C programming language (col. 2, line 65). It would have been obvious to one having an ordinary skill in the art at the time the invention was made to recognize that in Szoke's approach, a compiler lists the external

call as unresolved external references (see at least col. 1, line 39-40). Every high-level programming language comes with a compiler.

Therefore, one of ordinary skill in the art would have been motivated to apply Szoke's approach to C programming language as disclose in Tatge's approach because C is one of the high level programming languages.

15. Claims 12, 15, 16, 41, 44, and 45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Szoke (United States Patent No.: 4,787,034), in view of Sexton et al. (United States Patent No.: US 6,434,685).

As per claims 12 and 41:

Szoke discloses:

- creating an executable object module that includes symbolic references to addresses in ones of a set of one or more separately compiled object modules (**see at least FIG. 1, “load module 100 (LM-1)” and “program P11”**);
- replacing the symbolic references with addresses to a loader subroutine (see at least FIG. 1, “**CALL ‘P21’ or CALL ‘P24’**”);
- executing the executable object module (**“when load module 100 is created (executed)”** col. 4, line 2), wherein executing includes, executing the loader subroutine to load one of the separately compiled object modules (**“when the statement CALL ‘P21’ in program P11 is thereafter executed, control is**

passed to the statement at entry point P21 in E-table 120" col. 4, line 3-5);

and

- executing the one of the separately compiled object modules ("linkage program 130 transfers control to the actual address of program P21 in load module 200. Program P21 CALLs programs P22 and P23, and then returns to program P11" col. 4, line 45-48).

Szoke does not explicitly disclose:

- wherein the executable object module includes a page-aligned code segment and a page-aligned data segment, and wherein the object module includes resolved internal code-to-data offsets.

However, Sexton discloses a method for paged memory management system within a runtime environment that solves the page-aligned problem. It would have been obvious to one having an ordinary skill in the art at the time the invention was made to combine Sexton's approach with Szoke to improve the performance. The combination is obvious because one of ordinary skill would have motivated to save memory and improve the performance of page associated memory management operation by perform page-aligned code segment and a page-aligned data segment (see in Sexton at least col. 12, line 1-8).

As per claims 15 and 44:

Szoke discloses:

- wherein the unresolved reference is a reference to a function call to a function included in one of the separately compiled object modules of the set (“Program P11 includes statements that CALL programs P21” col. 2, line 63-64).

As per claims 16 and 45:

Szoke does not explicitly disclose:

- wherein the unresolved reference is a reference to a variable defined within one of the separately compiled objects of the set.

However, it would have been obvious to one having an ordinary skill in the art at the time the invention was made to recognize that calls to program P21 or P24 is also including calls to variables within program P21 or P24 of module 200. One of ordinary skill in the art would have been motivated to call to variables within P21 or P24 of module 200 because calling external variables are also considered as external unresolved references that need to be resolved dynamically at the time the module 100 is executed.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Phillip H. Nguyen whose telephone number is (571) 270-1070. The examiner can normally be reached on Monday - Thursday 10:00 AM - 3:00 PM EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Wei Y. Zhen can be reached on (571) 272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

PN
7/10/2008

/Wei Zhen/
Supervisory Patent Examiner, Art Unit 2191